Atty. Docket No.
005313.00001

PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent of:

Marc D. Van Heyningen

U.S. Pat. App. No.: 09/782,593

Filing Date: February 12, 2001

Examiner: L. Son

Group Art Unit: 2135

For: METHOD AND APPARATUS FOR PROVIDING SECURE STREAMING DATA TRANSMISSION FACILITIES USING UNRELIABLE PROTOCOLS

DECLARATION UNDER 37 C.F.R. § 1.131

Commissioner for Patents
P.O. Box 1450,
Alexandria, Virginia 22313-1450

Sir:

I, Marc D. Van Heyningen, do hereby declare as follows:

1. I am named as the inventor in U.S. Patent Application No. 09/782,593, filed February 12, 2001. I am an employee of Aventail Corporation having a place of business at 808 Howell Street, Second Floor, Seattle, Washington 98101, and I am over the age of eighteen years.

2. I am advised that Exhibit A contains a true and correct copy of the claims currently pending in U.S. Patent Application No. 09/782,593, namely claims 1-22. I have read and understand these claims.

3. I also have been informed that the United States Patent and Trademark Office has rejected claims 4-7 and 10-22 as contained in Exhibit A based upon U.S. Patent Application No. 2002/0094085 A1 to Roberts. I understand that the Roberts patent claims an effective U.S. filing

date of January 16, 2001.

4. I have reviewed (a) the computer code file entitled "sslctx.c" (a printed copy of which is attached as Exhibit B), (b) the computer code file entitled "sslrec.c" (a printed copy of which is attached as Exhibit C), (c) the computer code file entitled "ciphers.c" (a printed copy of which is attached as Exhibit D), and (d) the computer code entitled "sslenv.c" (a printed copy of which is attached as Exhibit E). I created each of these files of computer code by revising existing computer code to implement the invention as described in my patent application and recited in claims 4-7 and 10-22 (Exhibit A). I have obtained each of these files of computer code from our company's records. The copy of each of these files of computer code has been changed to include line numbers for the code lines. The line numbers were added in Exhibit B through Exhibit E to make the discussion below easier to follow, i.e., so that the cited code line numbers below would be readily available with the attached copy of the code.

5. As will be described in more detail below, the attached computer code, when implemented on a computer system, allows one to encrypt and transmit data records between a first computer and a second computer using an unreliable communication protocol in the manner described in my patent application and recited in claims 4-7 and 10-22 (Exhibit A).

6. One feature of a claim in my patent application relates to encrypting and transmitting data records between a first computer and a second computer using an unreliable communication protocol wherein each data record is encrypted by incorporating a nonce and without reference to a previously transmitted data record (note, for example, claims 4-6).

7. Referencing the source code, portions of the code that implement functionality relating

to my invention typically reference the term "SSLoppy", the working name for the development of the invention when this code was created. This term refers to the features provided by the invention that allows Secure Sockets Layer (SSL) records to be dropped and reordered during communication without disruption, which thus offer a "sloppy" variation of the SSL communication technique. The role of the first computer is included in: sslrec.c, lines 326-336, where the computer code adds extra room to a record for the inclusion of an explicit nonce, lines 354-378, where the computer code generates the nonce randomly, uses it in place of the sequence number in computing the MAC, and includes it in the record being built, lines 406-410, where the computer code passes this nonce to the encryption function, and in file cipher.c, lines 252-257, where the computer code uses the passed nonce value as an initial vector (IV), if present. The role of the second computer in reading these records is described in: file sslrec.c, lines 450-455, where the code parses the nonce out of incoming records, lines 466-475, where the code passes this nonce to the decryption function, and in file cipher.c, lines 323-328, where the code uses the passed nonce as the initial vector (IV), if present.

8. Another feature of a claim in my patent application relates to encrypting and transmitting data records between a first computer and a second computer using an unreliable communication protocol wherein an indicator is embedded in each of the data records indicating that the data records are encrypted according to an encryption scheme that encrypts records without regard to any previously transmitted data records, and the second computer determines whether the indicator is present in each record and, in response to determining that the indicator is not present, processes each such record differently than if the indicator is set (note, for example, claim 7).

9. Referencing the source code, in file sslrec.c, the role of the first computer in sending its different form of data is included in: lines 326-336, where the software code allocates extra space in the record for the nonce, if necessary, lines 354-391, where the ssloppy flag is used to decide whether to generate the record differently, lines 406-410, where the same flag is used to decide whether to use the nonce as the initial vector (IV) for the symmetric cipher algorithm, and lines 424-428, where the flag is used to decide whether to increment the sequence number. The role of the second computer, in receiving a packet of a different form of data, is included in: lines 95-99, where the software code checks a flag in the incoming record to verify that it is only receiving the new type of reorderable data when expected, lines 221-222, where the software code uses the same flag to determine whether to increment the sequence number, lines 450-454, where the software code uses the same flag to determine whether to extract the nonce from the record, lines 466-480, where the software code uses the same flag to decide whether to use this nonce as the initial vector (IV) when decrypting the record, and lines 497-517, where the software code uses the same flag to decide whether to use the nonce in place of the sequence number when verifying the MAC.

10. Still another feature of a claim in my patent application relates to securely transmitting a plurality of data records between a client computer and a proxy server using an unreliable communication protocol, that include the steps of establishing a reliable connection between the client computer and the proxy server, exchanging encryption credentials between the client computer and the proxy server over the reliable connection generating a nonce for each of a plurality of data records, wherein each nonce comprises an initialization vector necessary to

decrypt a corresponding one of the plurality of data records, using the nonce to encrypt each of the plurality of data records and appending the nonce to each of the plurality of data records, transmitting the plurality of encrypted data records from the client computer to the proxy server using an unreliable communication protocol, and, in the proxy server, decrypting each of the plurality of encrypted data records using a corresponding nonce extracted from each data record and a previously shared encryption key (note claims 10-15).

11. The proxy server and client architecture described above was already a commercial shipping product from Aventail, Inc. by January 16, 2001. Prior to the enhancements offered by the implementation of the invention, the product utilized the unreliable communication protocol "User Data Protocol" (UDP) to relay datagram records over a communication medium without any cryptographic protection. Comments in the software code refer to this process as "UDP Naked." The addition of the invention replaced "UDP Naked" with the "SSLoppy" process (as described above) thereby allowing communications to be cryptographically protected. Referencing the file sslenv.c, lines 1056-1067, for transmitting a datagram with this SSLoppy feature enabled, the SSLoppy encryption feature is turned on for the processing of a record by calling SSLSetSloppyMode to 1, and, subsequent to processing of the record, resetting this value back to 0 (described in lines 1131, 1145, 1153, 1183, 1220, 1244, 1252, 1259, 1278, 1286, 1314, 1347, 1359, and 1397). This software code processes only designated datagrams using the SSLoppy encryption process, and provides for standard SSL processing of data records being exchanged via a reliable communication protocol. The underlying functionality for selecting the SSLoppy encryption process is in sslctx.c, lines 899-911, where this function call sets the ctx-

>ssloppy flag. This flag is used to read and write records in the mode documented in paragraph 9 above.

12. Still another feature of a claim in my patent application relates to a system for securely transmitting data using an unreliable protocol that includes a first computer comprising a communication protocol client function operable in conjunction with an application program to transmit data records securely using an unreliable protocol, and a second computer coupled to the first computer and comprising a communication protocol server function operable in conjunction with the communication protocol client function to receive data records securely using the unreliable communication protocol, wherein the communication protocol client function encrypts each data record using a nonce and an encryption key and appends the respective nonce to each of the encrypted data records, and wherein the communication protocol server function decrypts each of the data records using the respectively appended nonce and the encryption key (note claims 16-22).

13. Supporting code is the same as that from paragraph 11 above.

14. The "source code" software code existed in the form shown in Exhibits B through E prior to January 16, 2001. This computer code received unit testing and was used as a proof-of-concept in the development process. Using this software code, the invention performed in its intended manner prior to January 16, 2001, as described in our patent application and claims 4-7 and 10-22 (as shown in Exhibit A), and thus was actually reduced to practice prior to January 16, 2001.

U.S. Pat. App. No.: 09/782,593
Atty. Docket No.: 005313.00001

DECLARATION IN LIEU OF OATH

15. I further declare that all information stated herein based upon my own knowledge is true and that all information stated herein based on information and belief is believed to be true, and further that the statements made in this Declaration were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of this application or any patent issuing based on this application.

Date: By: _____
Marc Van Heyningen, Inventor